Generalized Formalization of Game Rules

Christiaan van de Sande Tanner Reese

September 22, 2020

1 Introduction

2 Rules

Definition 2.1. A rule (r) on sets A and B is denoted by $r \in \mathcal{R}(A, B)$, with $\lambda_r : A \to \mathcal{P}(B)$, $\mu_r : A \times B \to B$, $\phi_r : A \times B \to B$, such that $\mu_r(\phi_r(a, b), b) = a$ and $\phi_r(\mu_r(a, b), b) = a$.

The functions of a rule are responsible for replicating the mechanics of a game. The input set, A, is the set of all possible game states; the output set, B, is the set of all well formed moves that could apply to any game state. Then, λ maps a given game state to a set of legal moves on that game state; μ maps a game state and a move to a new game state, and ϕ reverts the change made by μ.

3 Properties of rules

Definition 3.1. Let $r \in \mathcal{R}(A, B)$. r is **repeatable** iff $A \subseteq B$.

Definition 3.2. Let $r \in \mathcal{R}(A, B)$ and $s \in \mathcal{R}(A, C)$. r and s are **independent** $(r \perp s)$ iff, for all $(a \in A,\ b \in B,\ c \in C)$, $\mu_r(\mu_s(a, c), b) = \mu_s(\mu_r(a, b), c)$.

4 Operations on Rules

Definition 4.1. Let $r \in \mathcal{R}(A, B)$ and s be the **reduction** of r. Then $s \in \mathcal{R}(A, A)$, where:

$$s = \overline{r} \tag{1}$$

$$\lambda_s(a) = \begin{cases} \varnothing & \lambda_r(a) = \varnothing \\ \{a\} & \lambda_r(a) \neq \varnothing \end{cases} \tag{2}$$

$$\mu_s(a, b) = a \tag{3}$$

$$\phi_s(a, b) = a. \tag{4}$$

Definition 4.2. Let $r \in \mathcal{R}(A, A)$ and s be the **negation** of r. Then $s \in \mathcal{R}(A, A)$, where:

$$s = \neg r \tag{5}$$

$$\lambda_s(a) = \begin{cases} \varnothing & \lambda_r(a) \neq \varnothing \\ \{a\} & \lambda_r(a) = \varnothing \end{cases} \tag{6}$$

$$\mu_s(a, b) = \mu_r(a, b) \tag{7}$$

$$\phi_s(a, b) = \phi_r(a, b). \tag{8}$$

Definition 4.3. Let $r \in \mathcal{R}(A, B)$ and $s \in \mathcal{R}(A, B)$ and t be the **union** of r and s. Then $t \in \mathcal{R}(A, B)$, where:

$$t = s \cup r \tag{9}$$

$$\lambda_t(a) = \lambda_r(a) \cup \lambda_s(a) \tag{10}$$

$$\mu_t(a, b) = \mu_r(a, b) = \mu_s(a, b) \tag{11}$$

$$\phi_t(a, b) = \phi_r(a, b) = \phi_s(a, b). \tag{12}$$

Definition 4.4. Let $r \in \mathcal{R}(A, B)$ and $s \in \mathcal{R}(A, B)$ and t be the **intersection** of r and s. Then $t \in \mathcal{R}(A, B)$, where:

$$t = s \cap r \tag{13}$$
$$\lambda_t(a) = \lambda_r(a) \cap \lambda_s(a) \tag{14}$$
$$\mu_t(a, b) = \mu_r(a, b) = \mu_s(a, b) \tag{15}$$
$$\phi_t(a, b) = \phi_r(a, b) = \phi_s(a, b). \tag{16}$$

Definition 4.5. Let $r \in \mathcal{R}(A, B)$ and $s \in \mathcal{R}(A, C)$ and t be the **independent product** of r and s. Then $t \in \mathcal{R}(A, B \times C)$, where:

$$t = s \times r \tag{17}$$
$$\lambda_t(a) = \lambda_r(a) \times \lambda_s(a) \tag{18}$$
$$\mu_t(a, (b, c)) = \mu_r(\mu_s(a, c), b) \tag{19}$$
$$\phi_t(a, (b, c)) = \phi_s(\phi_r(a, b), c). \tag{20}$$

Definition 4.6. Let $r \in \mathcal{R}(A, B)$ and $s \in \mathcal{R}(B, C)$ and t be the **dependent product** of r and s. Then $t \in \mathcal{R}(A, B \times C)$, where:

$$t = s \cdot r \tag{21}$$
$$\lambda_t(a) = \{(b, c) | b \in \lambda_r(a), c \in \lambda_s(b)\} \tag{22}$$
$$\mu_t(a, (b, c)) = \mu_r(\mu_s(a, c), b) \tag{23}$$
$$\phi_t(a, (b, c)) = \phi_s(\phi_r(a, b), c). \tag{24}$$

Definition 4.7. Let $r \in \mathcal{R}(B, B)$ and $s \in \mathcal{R}(A, B)$ and $t \in \mathcal{R}(A, C)$ and v be r **patterned** from s to t. Then, $v \in \mathcal{R}(A, B)$, where:

$$v = r|_s^t \tag{25}$$
$$\nu_v(b) = \begin{cases} \{b\} & \lambda_t(b) \neq \varnothing \\ \{b\} \cup (\widehat{\nu}_v \circ \lambda_r(x)) & \lambda_t(b) = \varnothing \end{cases} \tag{26}$$
$$\lambda_v(a) = \widehat{\nu}_v \circ \lambda_s(a) \tag{27}$$
$$\mu_v(a, b) = \mu_s(a, \mu_r(b, b)) \tag{28}$$
$$\phi_v(a, b) = \phi_s(a, \phi_r(b, b)) \tag{29}$$

5 Implementation

6 Implications

6.1 Game Creation

6.2 Game Analysis

7 Conclusion